Berkeley Engineering MAS-E

Robust Optimization and Applications

Module 3: Chance Programming

Laurent El Ghaoui

Berkeley Engineering

Outline

E 238: Robust Optimization and Applications

Module 3: Chance Programming

Part 1: Overview

Berkeley Engineering

Motivations

The robust LP framework makes *no assumptions* about the distribution of uncertainty, except about the *support* of the distribution. For example, the uncertain vector may be assumed to belong to $[-1, 1]^n$ but nothing else is assumed.

If the modeler is ready to make extra assumptions, such as (partial) knowledge about the mean, then the chance programming framework can help produce solutions that exploit the extra knowledge, leading to less conservative solutions.

Objectives for This Module

- Learn how to cope with *random* uncertainty in optimization problems
- Understand the principles of chance programming
- Learn about a few tractable models
- Understand how chance programming relates to robust optimization via the certainty equivalent principle

E 238: Robust Optimization and Applications

Module 3: Chance Programming

Part 2: Chance Programming Basics

Berkeley Engineering

Chance-Constrained Linear Program

Nominal problem:

$$\min_x \; c^T x \;:\; Ax \leq b,$$

with A a $m \times n$ matrix. Now assume that the coefficient matrix A is random, with a known distribution π.

The chance counterpart enforces the constraint in probability:

$$\min_x \; c^T x \;:\; \mathbf{Prob}_\pi\{A \;:\; Ax \leq b\} \geq 1 - \epsilon$$

where $\epsilon > 0$ is typically small.

Distributional Robustness

In many cases of interest, it is not possible to fully know the distribution π; instead, we may assume some partial information is known, such as:

- The support of the random variables
- The mean, and sometimes the covariance
- Some other property such as unimodality

Denote by $\mathcal{P}$ the corresponding class of distributions.

We then solve a robust counterpart:

$$\min_x \; c^T x \;:\; \min_{\pi \in \mathcal{P}} \mathbf{Prob}_\pi\{A \;:\; Ax \leq b\} \geq 1 - \epsilon.$$

Challenges in Chance Programming

Chance programs are typically hard to solve:

- Just computing the probability of a set with x fixed, and π known, is already hard in general. In the interesting case when $\epsilon << 1$, adopting a sampling approach would require a huge number of samples
- In the context of optimization we would in addition have to optimize over a probability constraint
- The issue becomes completely intractable when π is only partially known

A Scalar Chance Constraint

To simplify, in the sequel we focus on a scalar chance constraint, of the form

$$\mathbf{Prob}_{\pi}\{a \,:\, a^T x \leq b\} \geq 1 - \epsilon,$$

where $x \in \mathbb{R}^n$ and $b \in \mathbb{R}$, $\epsilon \in [0, 1/2)$ are given, and a is a random vector.

We will consider the following cases:

- a is Gaussian
- a has known mean and covariance matrix, but otherwise unknown distribution
- The components of a are independent, with known support in $[-1, 1]$.
- The same as above, with an extra Gaussian assumption and with uncertainty on the mean

E 238: Robust Optimization and Applications

Module 3: Chance Programming

Part 3: Gaussian Uncertainty

Berkeley Engineering

Scalar Chance Constraint, Gaussian Case: Setup

Assume that the vector a follows a Gaussian distribution with mean $\hat{a}$ and covariance matrix C.

Since C is positive semi-definite (every eigenvalue is real, non-negative), there exist a matrix R such that $C = RR^T$.

Uncertainty model:

$$a = \hat{a} + Ru,$$

where u is a normal Gaussian variable: $u \sim \mathcal{N}(0, I)$.

Deterministic Counterpart

The chance constraint

$$\textbf{Prob}\{a \,:\, a^T x \leq b\} \geq 1 - \epsilon$$

is equivalent to

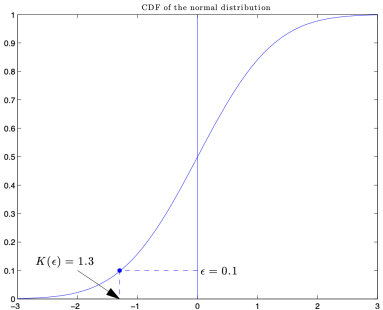
$$\hat{a}^T x + \rho(\epsilon)\|R^T x\|_2 \leq b,$$

where $\rho(\epsilon)$ is the negative of the inverse CDF of the normal distribution

$$\rho(\epsilon) = -\Phi^{-1}(\epsilon), \quad \Phi(\epsilon) := \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\epsilon} e^{-u^2/2} du.$$

Deterministic Counterpart

$$\textbf{Prob}\{a : a^T x \leq b\} \geq 1 - \epsilon \iff \hat{a}^T x + \rho(\epsilon)\|R^T x\|_2 \leq b.$$



- Above is an second-order cone constraint when $\epsilon < 1/2$
- Approach can be used for any linear inequality with Gaussian random coefficients

Certainty Equivalent Principle

The chance constraint

$$\hat{a}^T x + \rho(\epsilon)\|R^T x\|_2 \leq b$$

is a robust counterpart

$$\forall\, a \in \mathcal{U} \;:\; a^T x \leq b,$$

where $\mathcal{U}$ is an ellipsoid:

$$\mathcal{U} := \{\hat{a} + Ru \;:\; \|u\|_2 \leq \rho(\epsilon)\}$$

Interpretation

The chance constraint

$$\hat{a}^T x + \rho(\epsilon)\|R^T x\|_2 \leq b$$

makes a lot of sense:

- The constraint implies that $\hat{a}^T x \leq b$: we satisfy the constraint in expectation ...
- ... The term $\|R^T x\|_2$ is a risk premium that we have to "pay"; it makes the constraint more difficult to satisfy as the risk parameter $\rho(\epsilon)$ grows, *i.e.* ϵ decreases
- The term $\|R^T x\|_2 = \sqrt{x^T C x}$ is nothing else than the *standard deviation* of the scalar random variable $a^T x$

E 238: Robust Optimization and Applications

Module 3: Chance Programming

Part 4: Distributional Robustness

Berkeley Engineering

Principles of Distributional Robustness

We now assume that the random coefficient vector $a \in \mathbb{R}^n$ follows a distribution π that is only known to belong to a given class of distributions $\mathcal{P}$.

We now consider a *robust chance constraint:*

$$\forall\, \pi \in \mathcal{P} \;:\; \mathbf{Prob}_\pi\{a \;:\; a^T x \leq b\} \geq 1 - \epsilon.$$

Known Mean and Covariance Matrix

We assume that the vector a follows a (possibly non-Gaussian) distribution π with given mean $\hat{a}$ and covariance matrix C. Otherwise, π is unknown.

Uncertainty model:

$$a = \hat{a} + Ru,$$

where u is a zero-random variable with covariance matrix equal to the identity matrix I.

One-Sided Chebyschev Inequality

The (one-sided) Chebyschev inequality allows to bound the chance constraint using the mean and covariance information only.

The inequality states that, if ξ is a scalar random variable with mean $\hat{\xi}$ and standard deviation σ, and $\lambda \geq 0$, then

$$\textbf{Prob}\left\{\xi - \hat{\xi} > \lambda\right\} \leq \frac{\sigma^2}{\lambda^2 + \sigma^2}.$$

Deterministic Counterpart

Apply the inequality to the random variable $\xi = a^T x$. Since $\lambda := b - \hat{x}^T x \geq 0$, we obtain the robust chance constraint:

$$\hat{a}^T x + \rho(\epsilon)\|Rx\|_2 \leq b$$

where now the risk parameter is

$$\rho(\epsilon) := \sqrt{\frac{1-\epsilon}{\epsilon}}.$$

- This is similar to the Gaussian case
- The risk parameter is higher (more conservative)

Independent Variables, Box Support

Uncertainty model:

$$a = \hat{a} + Ru,$$

where u is a random variable with *independent* coordinates, all taking values in $[-1, 1]$.

We denote by $\mathcal{P}$ the set of corresponding distributions π of a.

Deterministic Counterpart

$$\forall\, \pi \in \mathcal{P} \;:\; \mathbf{Prob}_\pi\{a \;:\; a^T x \le b\} \ge 1-\epsilon \iff \hat{a}^T x + \rho(\epsilon)\|R^T x\|_2 \le b,$$

where

$$\rho(\epsilon) := \sqrt{2\log(1/\epsilon)}.$$

Proof: use the fact that, for any zero-mean vector random variable u with independent components and support in $[-1,1]$, deterministic n-vector z and scalar Ω:

$$\mathbf{Prob}\{u \;:\; u^T z \ge \Omega\|z\|_2\} \le \exp(-\Omega^2/2).$$

Certainty Equivalence

The chance constraint

$$\hat{a}^T x + \rho(\epsilon)\|R^T x\|_2 \leq b,$$

is the same as a robust counterpart, with uncertainty set

$$\mathcal{U} := \{\hat{a} + Ru \ : \ \|u\|_2 \leq \rho(\epsilon)\}$$

Discussion

- We could have obtained 100% reliability by enforcing the robust constraint over the entire support of the random variable $u \in B_1 := [-1, 1]^n$
- Instead we are enforcing the robust constraint only on the *much smaller* set $B_2 := \{u \ : \ \|u\|_2 \leq \rho(\epsilon)\}$. The ratio of the volumes $\mathbf{vol}(B_2)/\,\mathbf{vol}(B_1)$ goes very quickly to zero as the dimension of u grows
- One could instead try to enforce the robust counterpart on a set large enough to have $1 - \epsilon$ probability. This would still lead to an incomparably larger set, and a much more conservative condition
- If we choose the distribution π to be uniform on $\{-1, 1\}^n$, for $n \geq 28$, the probability of set $\mathcal{U}$ defined before is zero!

Independent Gaussians With Interval Mean Information

Uncertainty model:

$$a = \hat{a} + v + Ru,$$

Where

- $u \in \mathbb{R}^p$ is a zero-mean random variable with *independent* coordinates; for each i, variable u_i is a Gaussian random variable, with given variance σ_i
- The uncertainty on the mean, v, is deterministic, with $\|v\|_\infty \leq \epsilon$

We denote by $\mathcal{P}$ the set of corresponding distributions π of a.

Deterministic Counterpart

Robust chance constraint:

$$\forall\, \pi \in \mathcal{P} \;:\; \underset{\pi}{\mathbf{Prob}}\{a \;:\; a^T x \leq b\} \geq 1 - \epsilon \Longleftrightarrow \hat{a}^T x + \epsilon \|x\|_1 + \rho(\epsilon)\|R^T x\|_2 \leq b.$$

This is the same as a robust counterpart, with uncertainty set

$$\mathcal{U} := \{\hat{a} + v + Ru \;:\; \|v\|_\infty \leq \epsilon, \;\; \|u\|_2 \leq \rho(\epsilon)\}\,.$$

E 238: Robust Optimization and Applications

Module 3: Chance Programming

Part 5: Example: Investment Problem

Berkeley Engineering

Example: Investment Problem

We have several assets including cash (or, risk-free asset) and stocks, and denote by $r \in \mathbb{R}^{n+1}$ the vector of returns over the investment period, with r_{n+1} being the return of cash or a "risk-free" asset (say, $r_{n+1} = 1.05$ for a 5 % CD).

How should we distribute some sum (say, on thousand dollars) over these assets, where we take the returns to be a random variable?

Uncertainty model: We assume that the returns are random, independent, of the form $r_i = \hat{r}_i + \sigma_i u_i$, where $u \in [-1, 1]^{n+1}$ is random, zero-mean, with independent components. Let $\mathcal{P}$ be the corresponding class of distribution π of vector r.

Portfolio Optimization Model

We invest amounts $x \in \mathbb{R}^{n+1}$, such that $\mathbf{1}^T x = 1$ and $x \geq 0$. The total return of the portfolio is given by

$$R(x) := \sum_{i=1}^{n} r_i x_i.$$

We seek to solve the problem:

$$\max_x \ t \text{ subject to } \forall\, \pi \in \mathcal{P} \ : \ \mathbf{Prob}_\pi\{R(x) \geq t\} \geq 1 - \epsilon.$$

We are in effect minimizing the "value at risk", which is the largest t such that $R(x) < t$ has probability less than ϵ.

Experiment

In our experiment we assume that $n = 200$, and

$$\hat{r}_i = 1.05 + 0.3(200 - i)/199, \quad \sigma_i = 0.05 + 0.6(200 - i)/199, \text{ with } \sigma_{n+1} = 0.$$

Let us compare two models:

- A first model is when we use a purely robust counterpart using only support information: $u \in [-1, 1]^{n+1}$
- A second model is the chance counterpart described in page 24, with $R = \mathbf{diag}(\sigma)$

We use the reliability parameter $\epsilon = 0.005$.

Results

- Using the first model, we get $x = (0, 0, \ldots, 1)$: a purely robust approach leads to the very conservative investment of putting everything in the risk-free asset; the worst-case return is the risk-free return, 1.05
- The second model gives a worst-case return of 1.12 % with a 0.5 % chance of not getting this return

We thus observe that chance constraints allow to make the robustness condition much less conservative, at the expense of a very slight increase in risk.

E 238: Robust Optimization and Applications

Module 3: Chance Programming

Part 6: Example: Imbalanced Classification

Berkeley Engineering

Imbalanced Classification

Support vector machine (SVM) model for classification (see Module 2):

$$\min_{w,b} \frac{1}{m} \sum_{i=1}^{m} \max(1 - y_i(w^T \hat{x}_i + b), 0),$$

where

- Data points: $x_i \in \mathbb{R}^n$, $i = 1, \ldots, n$
- Labels: $y_i \in \{-1, 1\}$, $i = 1, \ldots, m$

In many applications, such as fraud detection, the positive class is much smaller than the negative one, leading to optimal classifiers that are simply ignoring the positive examples.

Cost-Sensitive Approach

The cost-sensitive approach simply gives more weight to the positive class (corresponding to indices $i \in \mathcal{I}_+$):

$$\min_{w,b} \ \theta L_+(w,b) + (1-\theta) L_-(w,b)$$

where $\theta \in [0,1]$ is a parameter, say $\theta = \frac{|\mathcal{I}_+|}{m}$, and

$$L_\pm(w,b) := \sum_{i \in \mathcal{I}_\pm} \max(1 - y_i(w^T \hat{x}_i + b), 0).$$

Chance Programming Approach

Uncertainty model for the negative class: we assume that negatively labelled point follow some distribution π that has given mean $\hat{x}$ and covariance matrix C. Let $\mathcal{P}$ be the corresponding class of distributions.

Model: We insist on having no training error on the positive class, and limit the chance that negative points are mis-labelled:

$$\begin{aligned} \min_{w,b,\epsilon} \ \epsilon \ : \quad & \forall\, i \in \mathcal{I}_+ \ : \ w^T \hat{x}_i + b \geq 0, \\ & \forall\, \pi \in \mathcal{P} \ : \ \mathbf{Prob}_\pi \left\{ x \ : \ w^T x + b \leq 0 \right\} \geq 1 - \epsilon. \end{aligned}$$

Certainty Equivalent Model

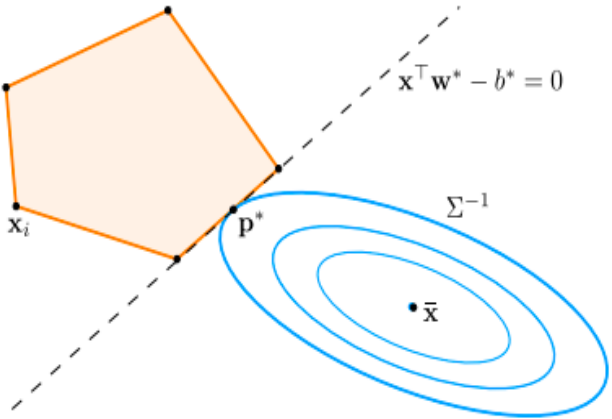
Problem writes

$$\max_{w,b,\kappa} \ \kappa \ : \ \ \forall \, i \in \mathcal{I}_+ \ : \ w^T \hat{x}_i + b \geq 0, \ \ w^T \hat{x} + b + \kappa \sqrt{w^T C w} \leq 0.$$

Exploiting the homogeneity in w, b, we can impose without loss of generality that $\kappa \sqrt{w^T C w} = 1$, and write the above as the QP

$$\min_{w,b} \ w^T C w \ : \ \ \forall \, i \in \mathcal{I}_+ \ : \ w^T \hat{x}_i + b \geq 0, \ \ w^T \hat{x} + b + 1 \leq 0.$$

Geometry



The negative class is modeled as a (partially known) distribution, while no error is tolerated for the positive class.

Results

Topic	This work	Cost-sensitive	Sampling
2	89.7 ± 1.0	89.9 ± 1.4	87.7 ± 1.2
9	96.1 ± 0.7	96.3 ± 0.8	94.1 ± 1.3
25	95.1 ± 0.8	94.3 ± 1.6	93.7 ± 1.2
33	96.0 ± 0.4	95.7 ± 0.6	93.9 ± 0.7
59	96.1 ± 0.4	95.9 ± 1.4	95.0 ± 0.6
84	96.9 ± 0.8	96.4 ± 1.5	96.3 ± 0.9

Topic	This work	Cost-sensitive	Speed-up
2	33	1088	33×
9	49	1451	29×
25	56	1211	21×
33	74	1788	24×
59	62	1299	21×
84	56	2056	36×

Results on a large news data set, with 1000 times more negative examples than positive ones. The chance approach has similar performance, with a training about $20 - 30$ times faster to solve.

E 238: Robust Optimization and Applications

Module 3: Chance Programming

Part 7: Summary

Berkeley Engineering

Summary

- Chance programming is a framework that relies on random uncertainty models and probabilistic constraints
- Distributional robustness model handle the case when the distribution is only partially known
- It can lead to tractable problems in the case of scalar chance constraints, and specific models such as Gaussian
- The corresponding problems often involve second-order cone constraints; the model can be interpreted as a (classical) robust counterpart, where the uncertainty set is typically much smaller than what a worst-case analysis (based on the support information only) would recommend